

Capitolo 4

4.1 Casi di Studio

4.1.1 Minimizza il Tempo di Missione

In questo caso, l'obiettivo è quello di minimizzare il tempo di missione per un gruppo di M velivoli e N target, che corrisponde a trovare l'allocazione ottima che consenta ai velivoli di visitare tutti i target e arrivare al rendez-vous nel minor tempo. L'allocazione è di tipo statico, nel senso che l'assegnamento rimane fisso durante tutta la missione. Non sono, inoltre, presenti vincoli temporali sul completamento della missione. La funzione di costo che si vuole minimizzare è la seguente:

$$\min \{T\}$$

$$T = \sum_{i=1}^M \max \{T_i\}$$

$$\forall \text{ velivolo } i \quad T_i = t_{i,Tg_i} + \sum_{j=2}^{N_i} t_{j-1,j}, t_{h,k} = \frac{d_{h,k}}{V_0}$$

Dove con N_i rappresentiamo l'insieme dei target che l'i-esimo velivolo deve visitare compreso il punto di rendez-vous.

Per semplicità, abbiamo usato una velocità costante, uguale per ogni membro del team di velivoli, in modo da poter considerare direttamente le distanze coperte in un generico percorso.

Graficamente la situazione può essere così rappresentata:

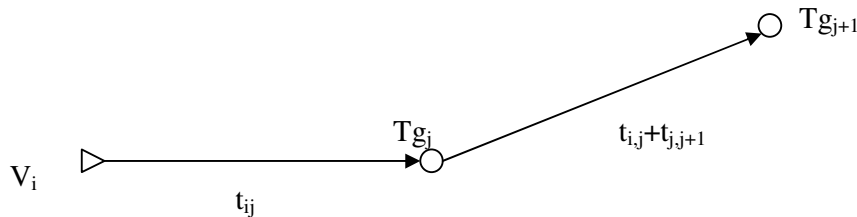


Figura 1: Rappresentazione Grafica della funzione di costo

La funzione di Fitness assume la seguente forma:

$$F = -T$$

Valori bassi della funzione di costo corrispondono a valori alti del fitness.

La soluzione utilizzata impone che ad ogni velivolo sia assegnato almeno un target.

Un possibile scenario è illustrato in Figura 2.

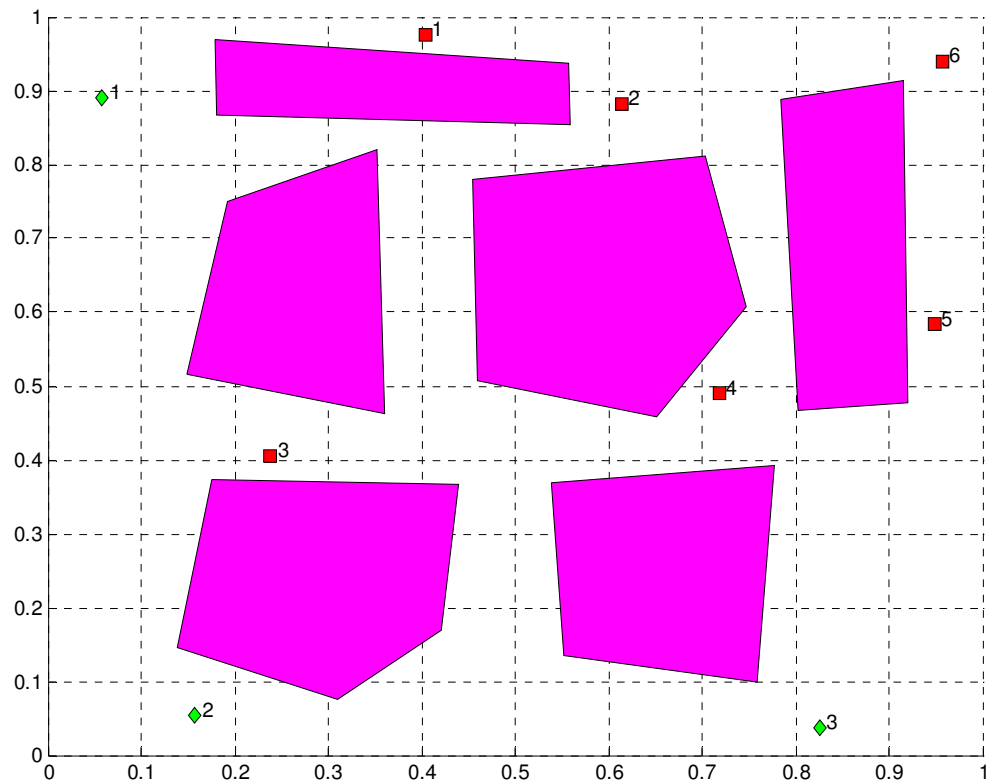


Figura 2: 3 velivoli su 5 target

I parametri utilizzati dal GA sono:

- Dimensione della popolazione iniziale pari a 100
- Numero di generazioni pari a 20

Il risultato della simulazione è mostrato in Figura 3:

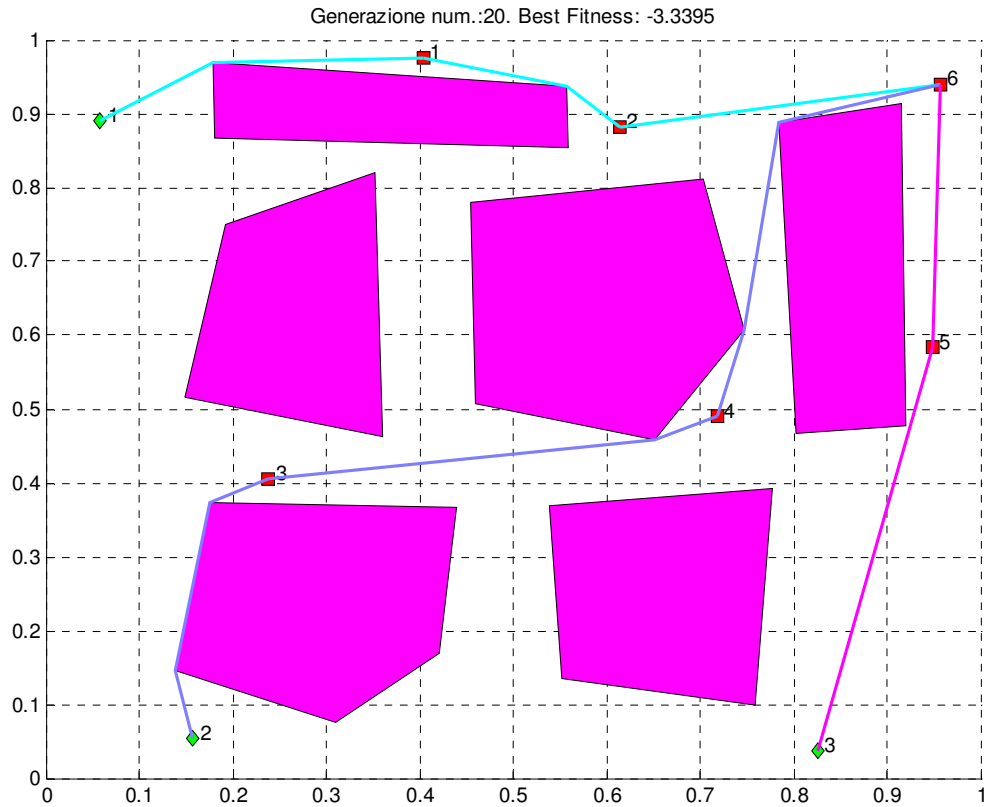


Figura 3: Assegnamento ottimo

Il tempo di calcolo richiesto è mediamente pari a 2,8 secondi.

L'andamento del fitness durante una generica prove assume l'andamento di Figura 4.

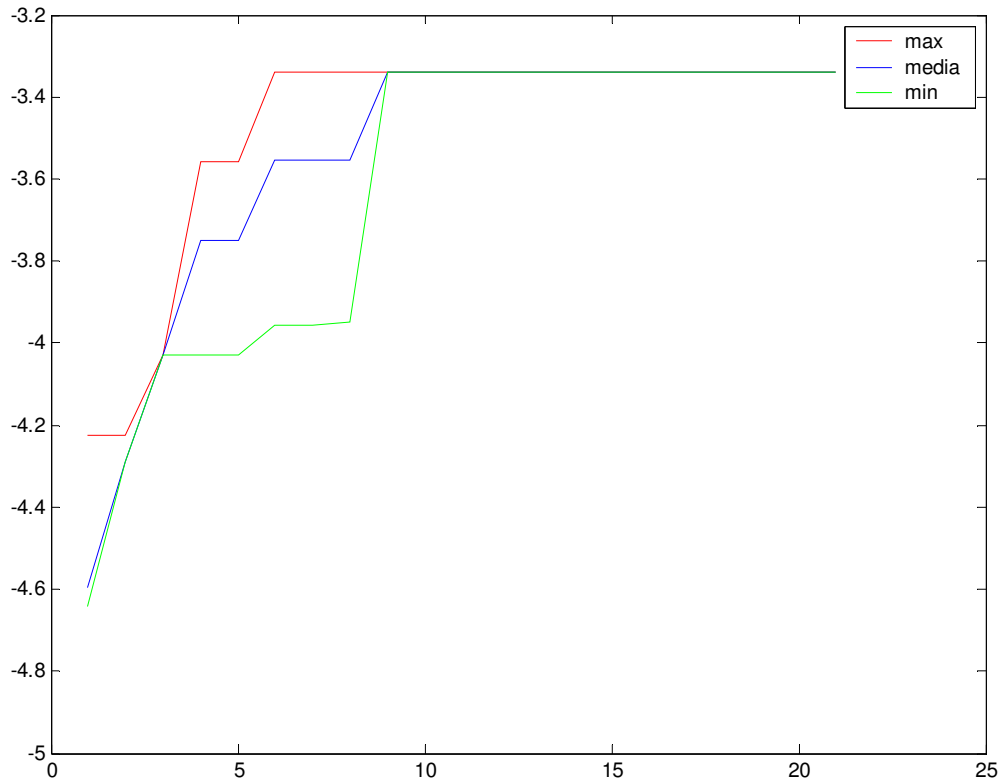


Figura 4: Andamento del Fitness nelle varie generazioni

La probabilità di convergenza su 100 prove risulta pari al 96%.

Si nota che all’aumentare del numero di generazioni la probabilità di convergenza aumenta.

Altre prove con diversi valori per la dimensione della popolazione e del numero di generazioni sono riassunte in Tabella 1.

Dimensione della popolazione	Numero di Generazioni	Tempo di Calcolo Medio	% di Convergenza su 100 prove
20	20	0.6	50

20	30	0.8	62
20	40	1	78
40	20	1	69
20	100	2	84
100	20	2.8	96

Tabella 1: Confronto tra varie condizioni

Consideriamo ora uno scenario più complesso con un team di 3 velivoli e 10 target come in Figura 5.

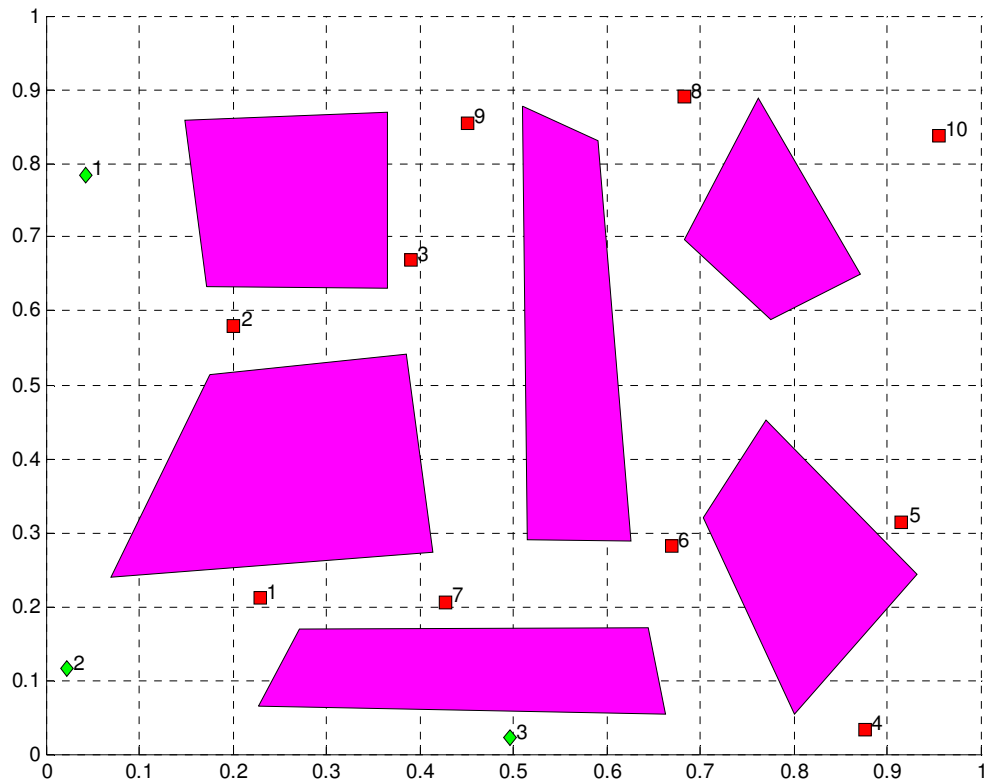


Figura 5: Secondo scenario

I parametri utilizzati dal GA sono:

- Popolazione iniziale di dimensione pari a 100
- Numero di generazioni pari a 100

Il risultato della simulazione è mostrato in Figura 6.

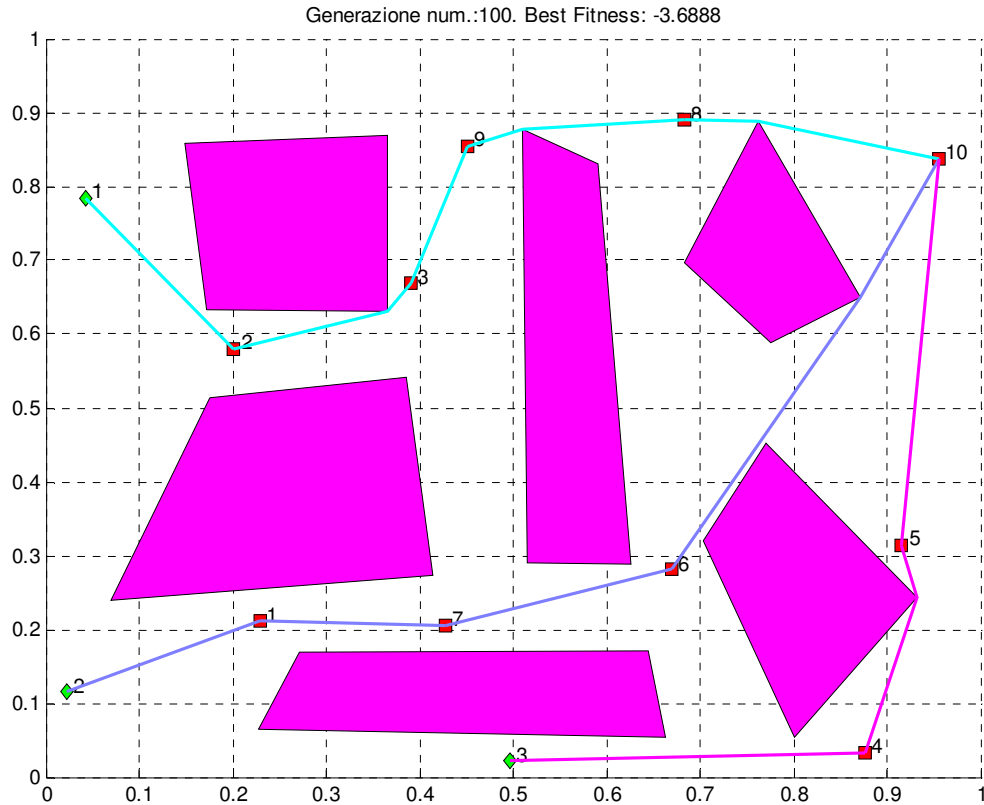


Figura 6: Allocazione ottima

Il tempo di calcolo richiesto è mediamente pari a 9 secondi.

La percentuale di convergenza su 100 prove è del 25%.

Altre prove con diversi valori per il numero di generazioni sono riassunte in Tabella

2.

Dimensione della popolazione	Numero di Generazioni	Tempo di Calcolo Medio	% di Convergenza su 100 prove
100	200	18	38
100	300	25	55

Tabella 2: Confronto tra varie condizioni

Si evince che all'aumentare della complessità dello scenario bisogna aumentare il numero di generazioni per ottenere una più alta percentuale di convergenza.

4.1.2 Minimizza il Tempo di Missione con vincoli

In questo caso l'obiettivo è quello di minimizzare il tempo di fine missione per un gruppo di M velivoli con vincoli sul tempo massimo di missione. Ogni velivolo deve arrivare al rendez_vous entro un tempo prestabilito. La funzione di costo da minimizzare è la seguente:

$$\min \{T\}$$

$$T = \sum_{i=1}^M \max \{T_i\}$$

$$\forall \text{ velivolo } i \quad T_i = t_{i, TG_1} + \sum_{j=2}^{N_i} t_{j-1, j} \leq T_{\max, i}, t_{h, k} = \frac{d_{h, k}}{V_0}$$

Dove con N_i rappresentiamo l'insieme dei target che l' i -esimo velivolo deve visitare compreso il punto di rendez-vous.

La funzione di Fitness utilizzata assume la seguente forma:

$$\forall \text{ velivolo } i: \sum_{j=1}^{N_i} \begin{cases} k * (T_{\max,i} - (t_{i,1} + \sum_{k=1}^j t_{j,k})) & \text{se } T_{\max,i} - (t_{i,1} + \sum_{k=1}^j t_{j,k}) < 0 \\ (T_{\max,i} - (t_{i,1} + \sum_{k=1}^j t_{j,k})) * \exp(-(T_{\max,i} - (t_{i,1} + \sum_{k=1}^j t_{j,k}))^2) & \text{altrimenti} \end{cases}$$

Dove per la costante k utilizziamo un valore alto in modo da eliminare le allocazioni che portano i velivoli a superare i limiti di tempo imposti. La funzione riesce a modellare l'incertezza sul futuro, nel senso che, poiché l'andamento della funzione (vedi Figura 7) assume il massimo in una posizione diversa da zero (situazione in cui un velivolo i completa la missione esattamente in $T_{\max,i}$), la funzione di Fitness porta a scegliere come ottima una allocazione che consente al velivolo di "conservare" del tempo per visitare in futuro altri target.

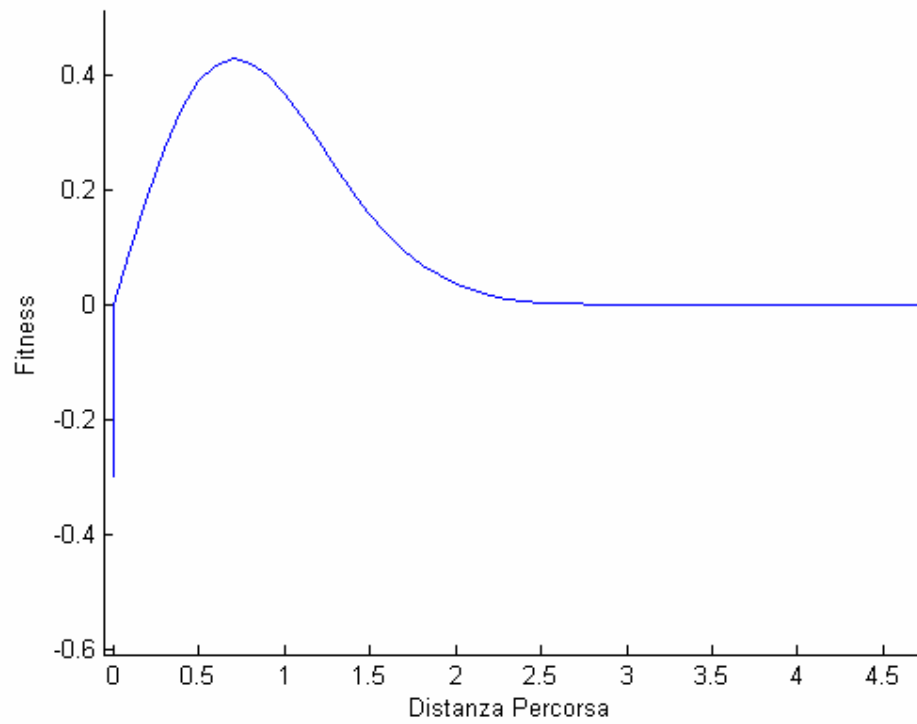


Figura 7: Andamento del fitness per un velivolo e un target

Consideriamo il seguente scenario mostrato in Figura 7.

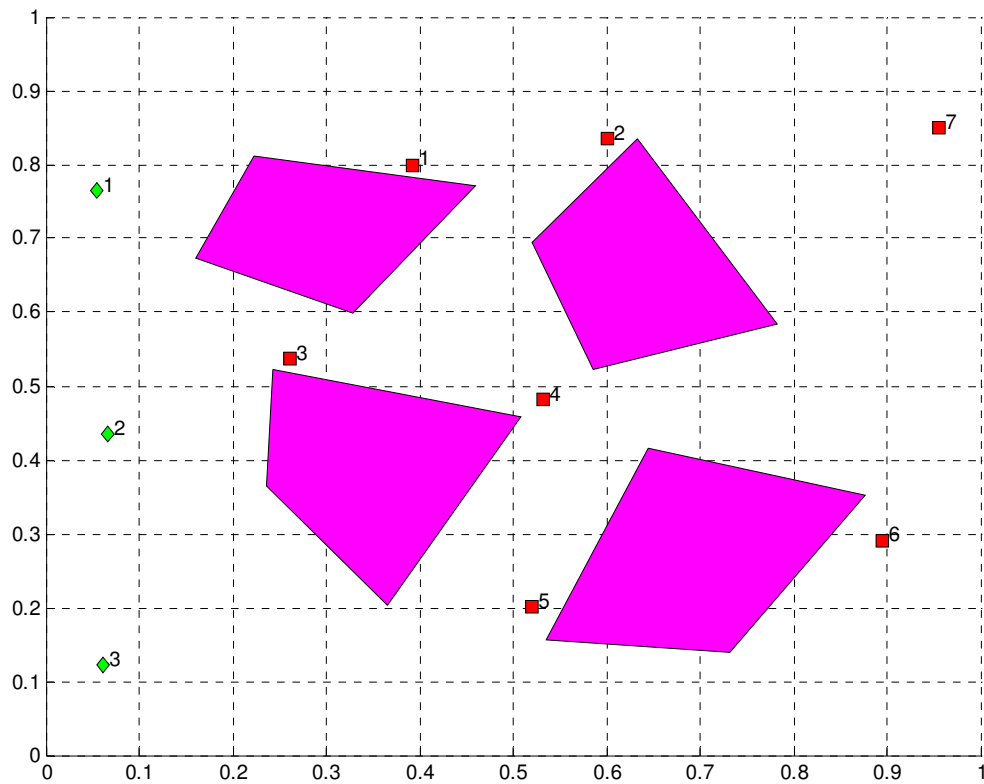


Figura 8: 3 velivoli su 6 target

I parametri utilizzati dal GA sono:

- Dimensione della popolazione iniziale pari a 100
- Numero di generazioni pari a 100
- I tempi massimi risultano [1.1 1.2 1.3]

Il risultato della simulazione è mostrato in Figura 8:

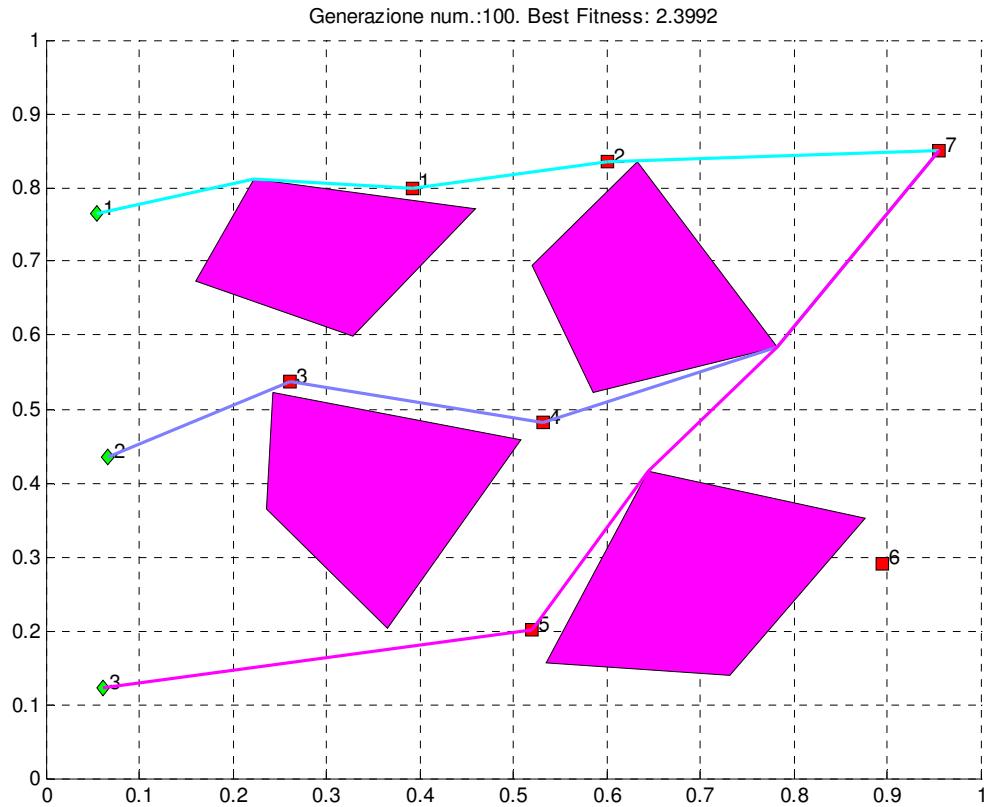


Figura 9: Allocazione ottima

Il tempo di calcolo richiesto è mediamente pari a 13 secondi.

L'andamento del fitness durante una generica prove assume l'andamento di Figura 9.

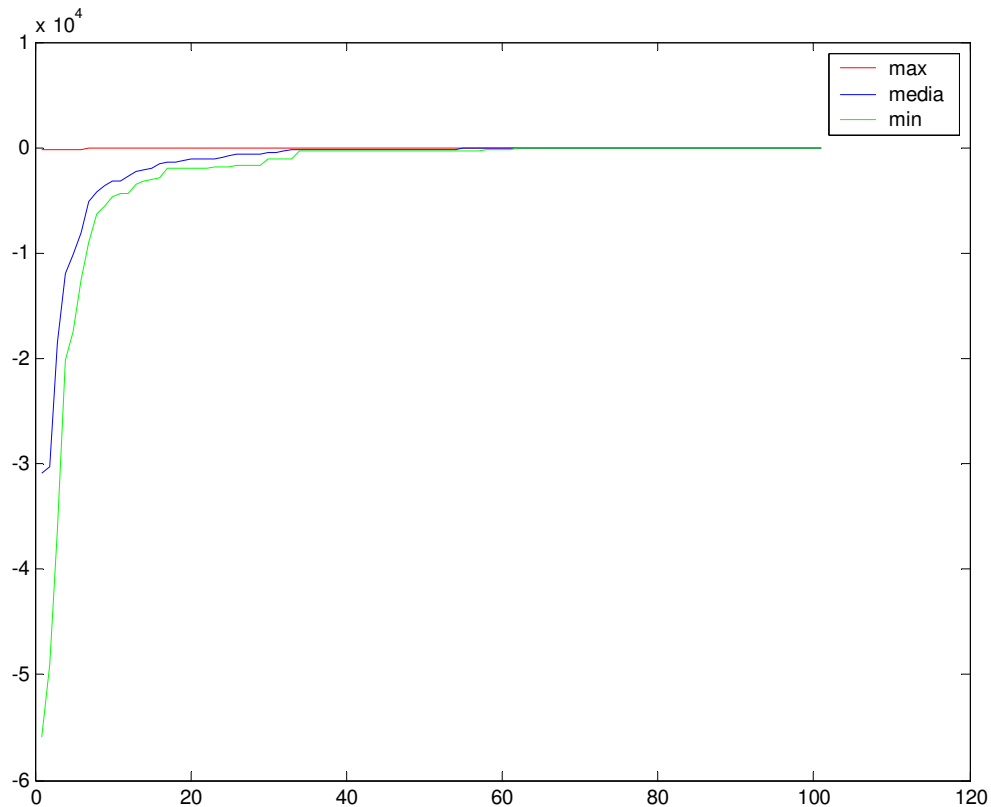


Figura 10: Andamento del Fitness

La probabilità di convergenza su 100 prove è del 96%.

Come si vede dalla Figura 9, l'assegnamento ottimo impone che il target 6 non venga visitato; infatti i tempi residui, che risultano [0.189 0.1153 0.0513], non consentono di visitare tutti i target. Calcolando i tempi finali di missione per un assegnamento che prevede la visita del target 6 otteniamo per il velivolo 3 il seguente risultato, $T_{\text{missione}}=1.494$, che supera il tempo massimo assegnatoli. Modificando il tempo massimo di missione del velivolo 3, quindi utilizzando il seguente vettore di tempi

massimi [1.1 1.2 1.5], l’algoritmo assegna anche il target 6. L’assegnamento ottimo in questo caso è mostrato in Figura 10.

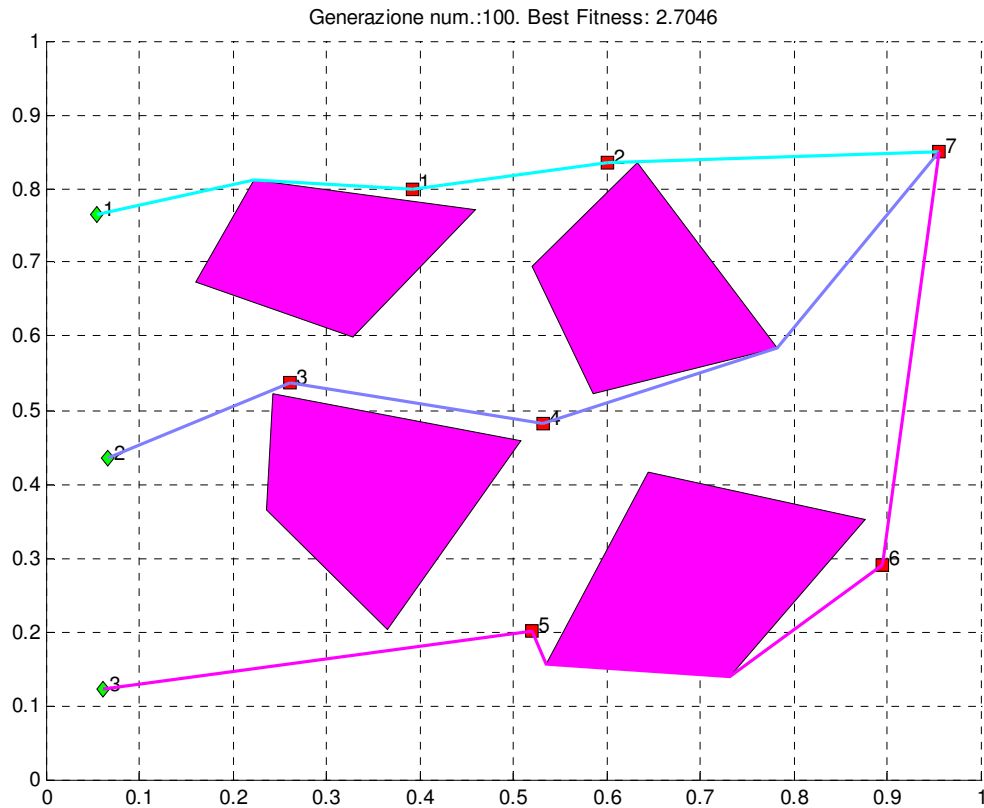


Figura 11: Allocazione ottima con Tempi massimi [1.1 1.2 1.5]

4.1.3 Velivoli con Attitudine al Rischio

In questo caso, cerchiamo di modellare una situazione in cui i target siano di differenti tipi e possano introdurre diversi Gradi di Pericolosità (Score), ossia

rappresentare minacce più o meno grandi; ad esempio target che rappresentano, in una ipotetica missione di combattimento, postazioni radar oppure postazioni lanciamissili. Di conseguenza, bisognerà utilizzare velivoli con caratteristiche e attitudini diverse che discriminino la possibilità di assegnare loro particolari obiettivi; chiameremo Capacità Operativa l'attitudine di un velivolo ad affrontare particolari obiettivi in funzione del loro grado di pericolosità.

$$\begin{aligned} \forall \text{ velivolo } i \text{ } Cop_i &\in [0 \quad 1] \\ \forall \text{ obiettivo } j \text{ } Sc_j &\in [0 \quad 1] \end{aligned}$$

La finalità della missione è quella di impedire ai velivoli con poca capacità operativa di affrontare target con alto score; cercando di assegnare, comunque, ai velivoli il numero maggiore di target con il vincolo che la somma dei loro score non superi la capacità operativa del velivolo a cui sono assegnati; in più, la somma delle capacità operative deve essere maggiore o uguale alla somma degli score. Inoltre si cerca di minimizzare anche il tempo di missione. La funzione di costo da minimizzare può essere così formulata:

$$\begin{aligned} \min\{Cs\} \\ Cs &= \sum_{i=1}^M (Cs_i) \\ \forall \text{ velivolo } i \text{ } Cs_i &= (Cop_i - \sum_{j=1}^{N_i} Sc_j) / (t_{i,Tg_i} + \sum_{j=2}^{N_i+RV} t_{j-1,j}) \end{aligned}$$

Dove con N_i rappresentiamo l'insieme dei target che l'i-esimo velivolo deve visitare escluso il punto di rendez-vous.

Graficamente la situazione può essere rappresentata come:

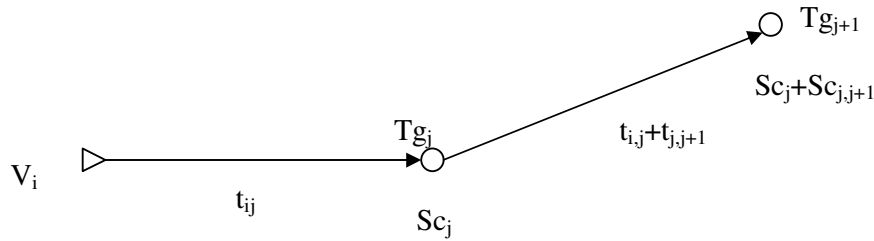


Figura 12: Rappresentazione grafica della funzione di costo

La funzione di Fitness utilizzata assume la seguente forma:

$$\forall \text{ velivolo } i \quad f_i = \frac{\exp(-(\text{Cop}_i - \sum_{j=1}^{N_i} Sc_j)^2)}{(t_{i,Tg_i} + \sum_{j=2}^{N_i+RV} t_{j-1,j})}$$

Dalla formula si vede che la fitness di ogni velivolo assume valore massimo in corrispondenza di assegnamenti che annullano la capacità operativa del velivolo e minimizzano la distanza totale percorsa. Si impone che il punto di rendez-vous abbia score nullo.

Consideriamo lo scenario in Figura 12, con i seguenti parametri:

- Capacità Operativa pari a [1 1 1 1 1]
- Score dei target [1 1 1 0.2 0.4 0.4 0.2 0.3 0.4 0.1 0]
- Dimensione della popolazione pari a 100
- Numero di generazioni pari a 200

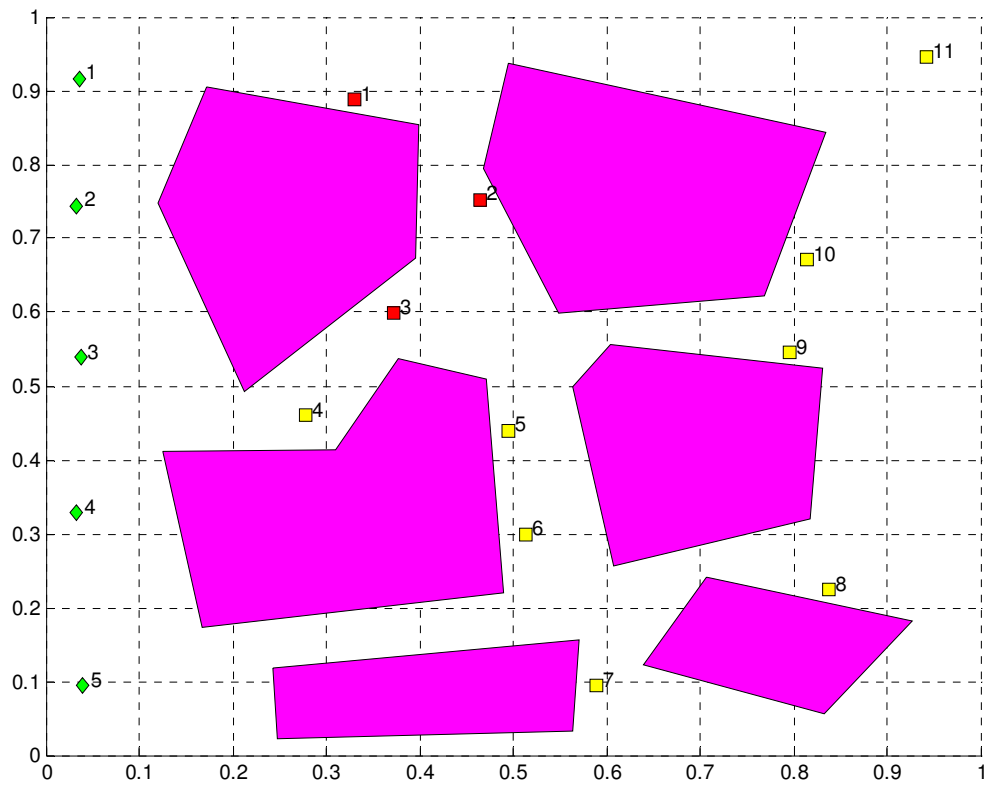


Figura 13: Possibile scenario

L'assegnamento ottimo è mostrato in Figura 13

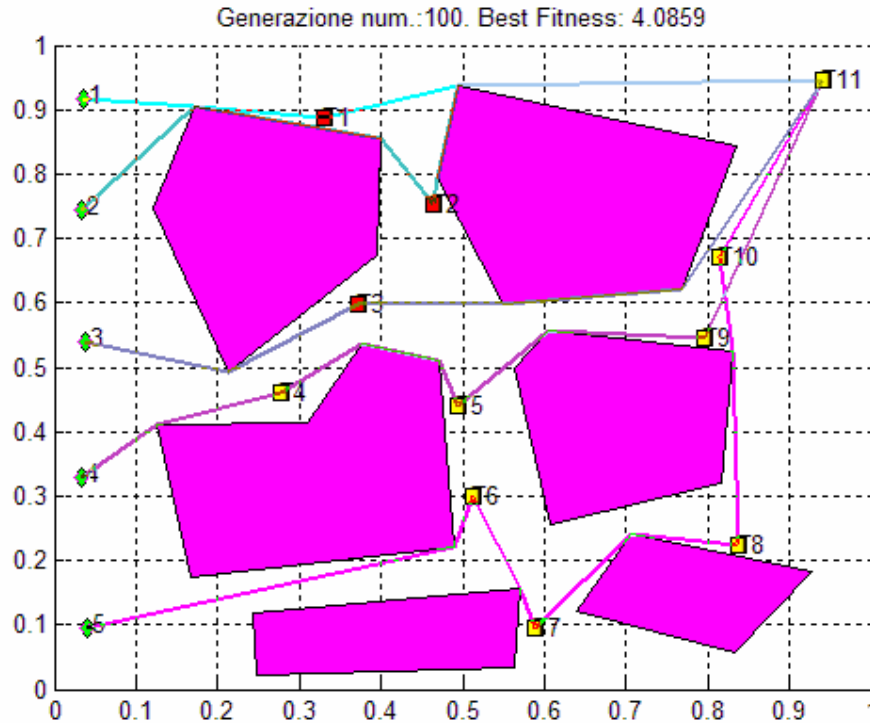


Figura 14: Allocazione ottima

Il tempo di calcolo richiesto è mediamente pari a 24 secondi; la percentuale di convergenza su 100 prove è del 50%.

4.1.4 Capacità operativa e Score variabili

Diversamente dal problema precedente, in cui le capacità operative e gli score sono fissi, analizziamo il caso in cui entrambi variano in funzione della distanza percorsa.

Le funzioni che descrivono le capacità e gli score risultano:

$$\begin{aligned}
 Cop_i &= Cop_{nom} * \exp(-(d)^2) \\
 Sc_j &= Sc_{nom} * \exp(-(d/0.1)^2)
 \end{aligned}$$

L'andamento, nel caso di una $Cop_{nom}=1$ e uno $SC_{nom}=0.4$, è mostrato in Figura 14

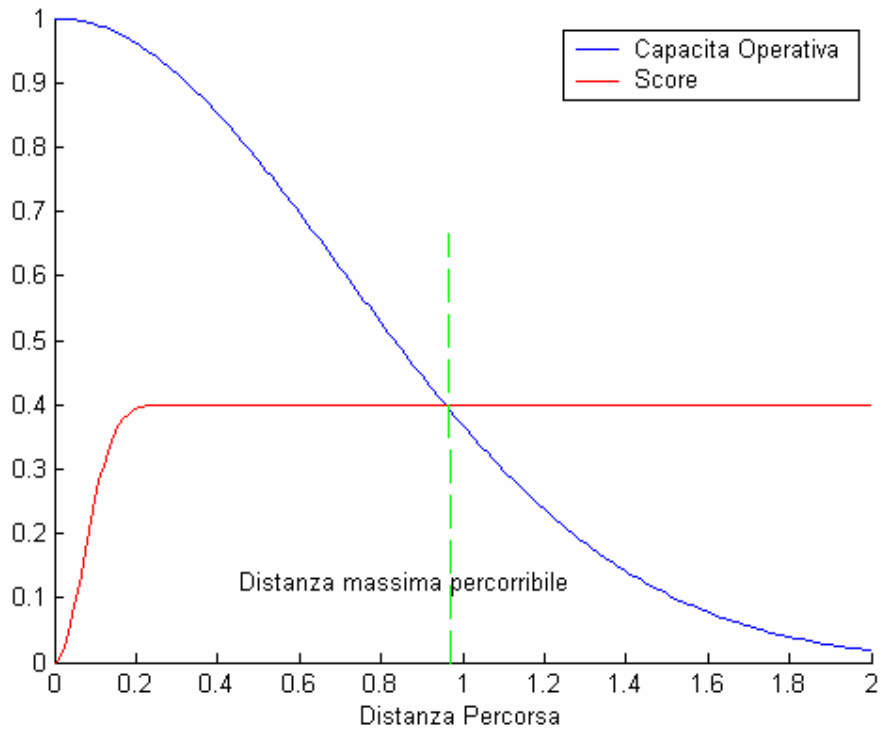


Figura 15: Andamento della capacità e dello score

In questo caso, un altro fattore discriminante, oltre alle capacità e agli score , è dato dal valore dello score del punto di rendez-vous. Infatti, maggiore è tale valore, minore risulta la distanza che il velivolo deve percorrere per raggiungere tale posizione; la distanza massima, che ogni velivolo può percorrere, è data dunque dall’intersezione tra la curva del generico velivolo e la curva dello score del rendez-vous. La funzione di Fitness utilizzata è la seguente:

$$\forall i: \sum_{j=1}^{N_i} \begin{cases} 100 * (Cop(t_{i,1} + \sum_{k=1}^j t_{j,k}) - Sc_j(t_{i,1} + \sum_{k=1}^j t_{j,k})) \text{ se negativo} \\ (Cop(t_{i,1} + \sum_{k=1}^j t_{j,k}) - Sc_j(t_{i,1} + \sum_{k=1}^j t_{j,k})) * \exp(- (Cop(t_{i,1} + \sum_{k=1}^j t_{j,k}) - Sc_j(t_{i,1} + \sum_{k=1}^j t_{j,k}))^2) \end{cases}$$

Dove con N_i rappresentiamo l’insieme dei target che l’i-esimo velivolo deve visitare compreso il punto di rendez-vous. Consideriamo il seguente scenario:

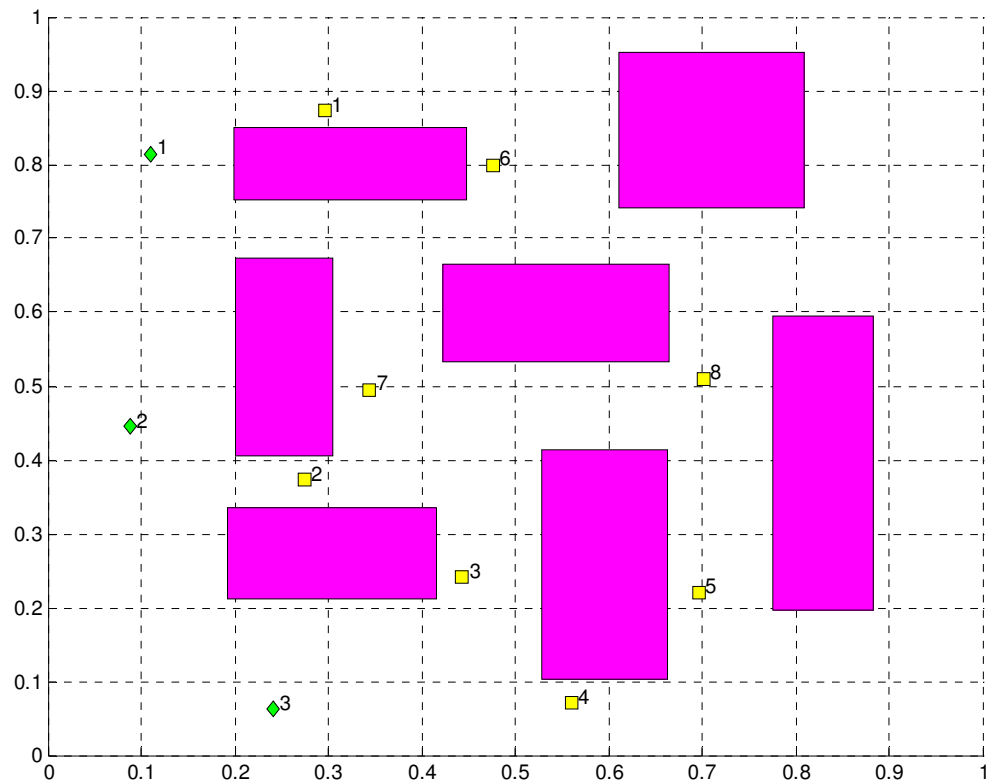


Figura 16: Possibile scenario

I parametri utilizzati risultano:

- Capacità Operativa pari a [0.9 0.9 0.9]
- Score dei target [0.8 0.4 0.2 0.3 0.1 0.1 0.2 0.1]
- Dimensione della popolazione pari a 100
- Numero di generazioni pari a 100

La soluzione ottima è mostrata in Figura 16

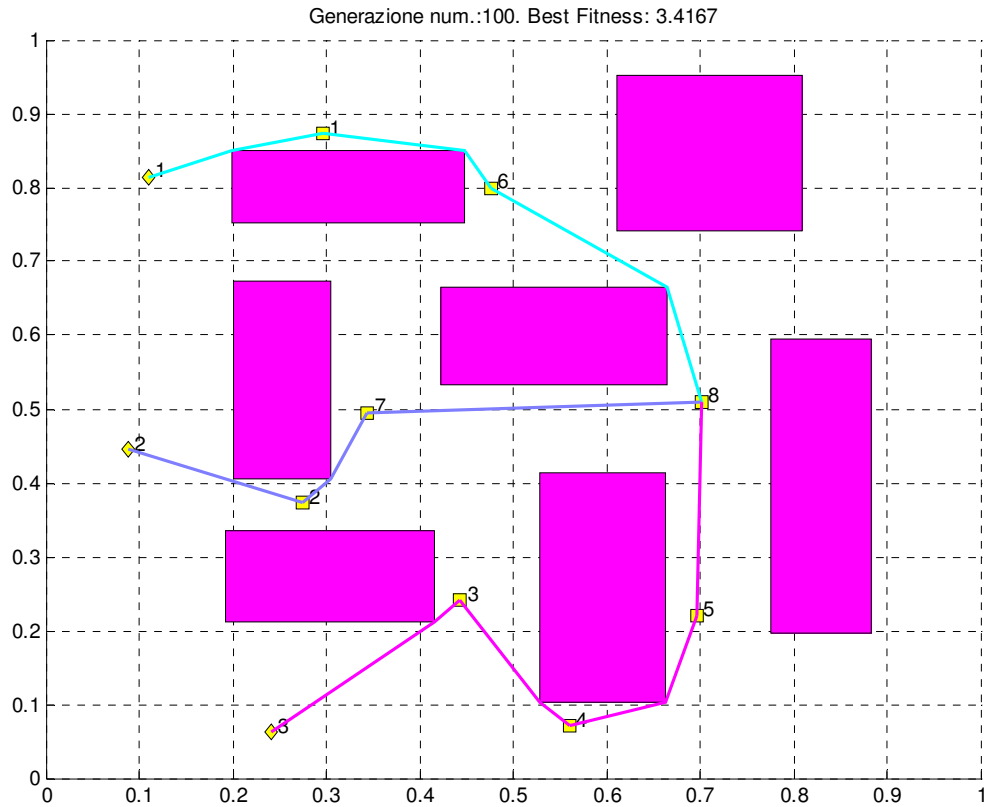


Figura 17: Allocazione ottima

Il tempo di calcolo è mediamente pari a 14 secondi; la percentuale di convergenza su 100 prove è del 40%.

Portando lo score nominale del rendez-vous a 0.4, l’allocazione precedente comporta che il velivolo 3 superi la distanza massima consentita per raggiungere il rendez-vous; l’algoritmo restituisce come soluzione ottima la seguente:

- Velivolo 1: target 1 -> target 6
- Velivolo 2: target 2 -> target 7
- Velivolo 3: target 4 -> target 5

4.1.5 Target con Score Esteso

Rispetto al problema precedente, in cui i target venivano considerati singolarmente, ora ogni target visitato contribuisce ad aumentare lo score del target successivo; quindi la capacità del singolo velivolo diminuisce oltre che in base alla distanza percorsa anche in base ai target visitati. La possibilità di compiere una missione, nel senso di visitare il maggior numero di target con i vincolo imposti, dipendono fortemente dall'andamento della funzione di capacità e dello score; infatti più la funzione di capacità decresce lentamente e la funzione di score cresce lentamente maggiori sono le possibilità di compiere la missione.

Le funzioni diventano:

$$\begin{aligned} Cop_i &= Cop_{nom} * \exp(-(d / \alpha)^2) \\ Sc_j &= Sc_{nom} * \exp(-(d / \beta)^2) \end{aligned}$$

Al variare dei parametri α e β , dovremmo trovare soluzioni che aumentino le capacità del team di velivoli di visitare più target. La funzione di Fitness è la seguente:

$$\forall i: \sum_{j=1}^{N_i} \left\{ \begin{array}{l} k * (Cop(t_{i,1} + \sum_{k=1}^j t_{j,k}) - \sum_{h=1}^j Sc_h(t_{i,1} + \sum_{k=1}^h t_{h,k})) \text{ se negativo} \\ (Cop(t_{i,1} + \sum_{k=1}^j t_{j,k}) - \sum_{h=1}^j Sc_h(t_{i,1} + \sum_{k=1}^h t_{h,k})) * \exp(-(Cop(t_{i,1} + \sum_{k=1}^j t_{j,k}) - \sum_{h=1}^j Sc_h(t_{i,1} + \sum_{k=1}^h t_{h,k}))^2) \end{array} \right.$$

Poiché si vuole minimizzare anche il tempo di missione la funzione viene poi divisa per la distanza percorsa dal singolo velivolo.

Consideriamo lo scenario in Figura 16. I parametri utilizzati risultano:

- Capacità Operativa pari a [1 1 1]

- Score dei target [0.8 0.4 0.2 0.3 0.1 0.1 0.2 0.1]
- Dimensione della popolazione pari a 100
- Numero di generazioni pari a 100

Possibili soluzioni al variare di α , con $\beta=0.1$ sono mostrate in Figura 18.

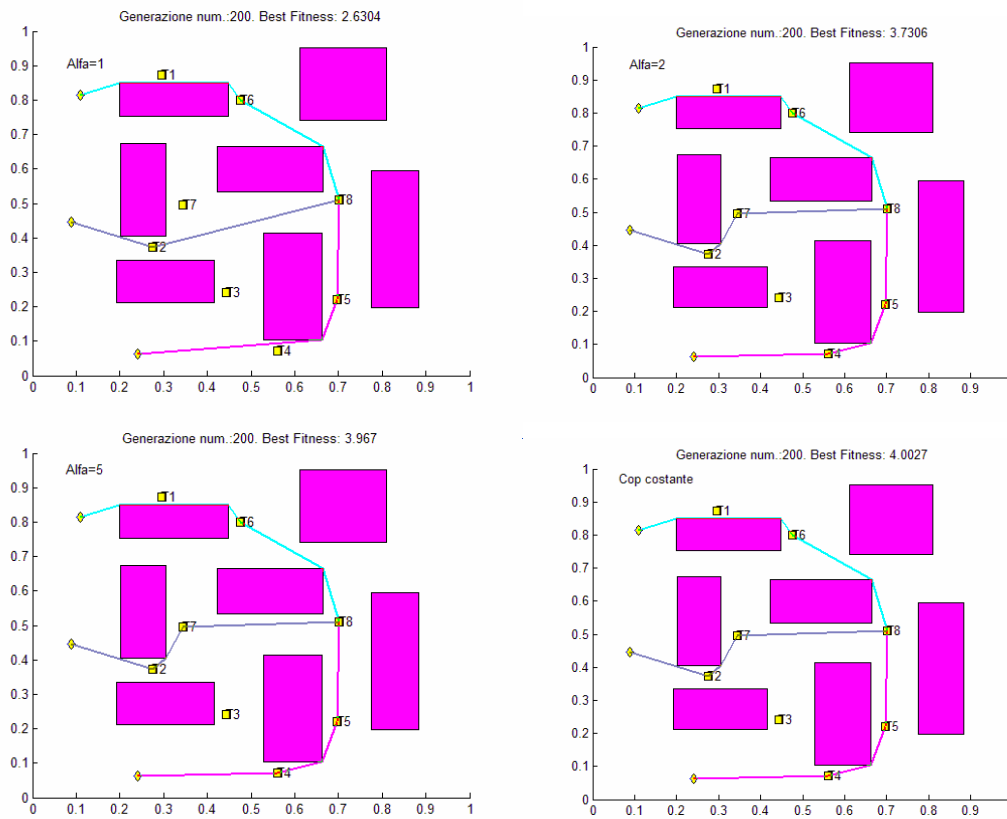


Figura 18: Soluzioni al variare di α

Le varie prove mostrano che la fitness cerca sempre di massimizzare la capacità residua dei velivoli. Considerando il caso di $\alpha=7$, l’algoritmo riesce ad allocare tutti i target; l’allocazione è la seguente:

- Velivolo 1: target 1 -> target 6
- Velivolo 2: target 2 -> target 7
- Velivolo 3: target 3 -> target 4 -> target 5

La funzione di fitness, comunque, restituisce un valore minore rispetto agli assegnamenti che si vedono in Figura 18 a partire dal caso con $\alpha=2$, questo perché le capacità residue sono minori. I valori delle capacità residue nei due casi è mostrato in

Tabella 3

	Velivolo 1	Velivolo 2	Velivolo 3
Assegnamento ottimo	0.7872	0.2972	0.4858
Assegnamento 2	0.0042	0.2972	0.2801

Tabella 3

Vediamo cosa succede al variare di β , con $\alpha=2$. I risultati sono mostrati in Figura 19.

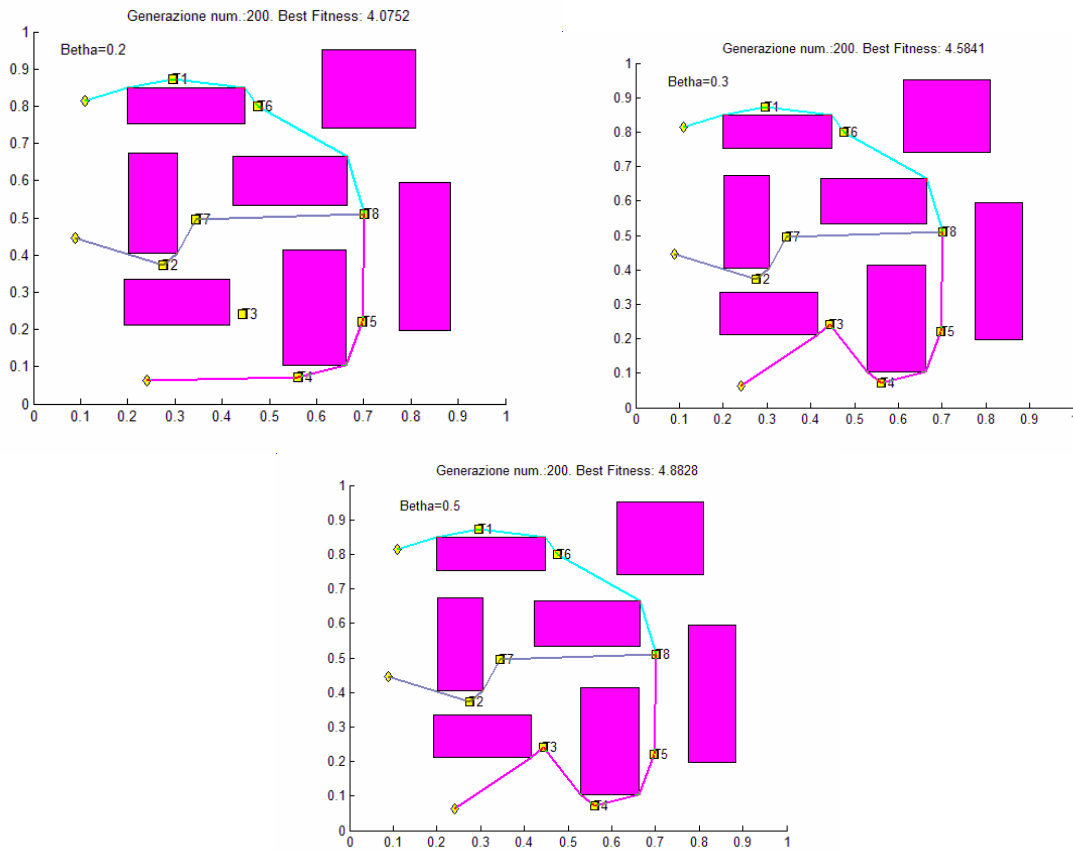


Figura 19: Soluzioni al variare di β

Le prove mostrano che variando in modo opportuno i valori di α e β si riesce ad aumentare la capacità del team di visitare più target massimizzando comunque le capacità residue dei singoli velivoli.

4.2 Simulazioni

Il passo successivo del lavoro di tesi, è stato quello di testare alcuni dei casi studiati simulando una possibile missione.

La dinamica dei velivoli è stata modellata usando un semplice integratore, mediante il seguente schema Simulink:

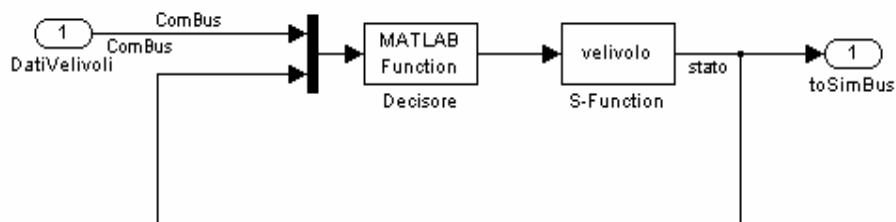


Figura 19: Schema del velivolo i-esimo

La maschera utilizzata tiene conto dei seguenti parametri

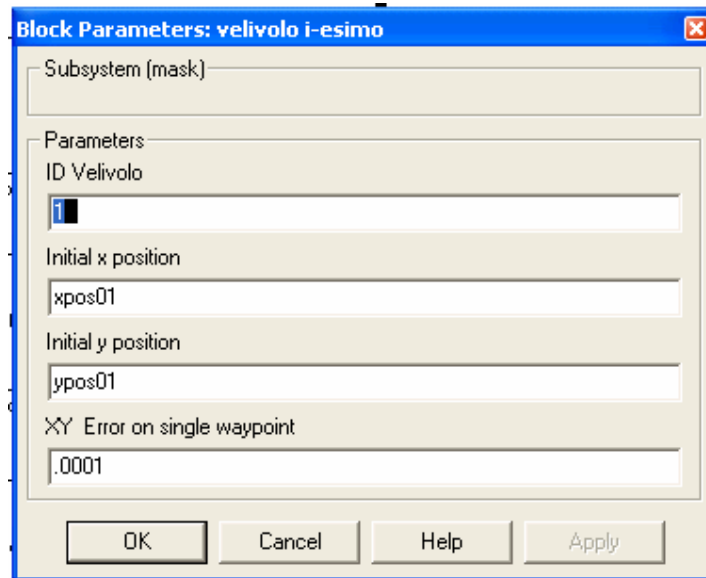


Figura 20: Parametri della maschera del velivolo

Al generico velivolo sono passati, inoltre, i vari waypoint, generati durante l'ottimizzazione, che costituiscono il percorso; mentre il decisore restituisce semplicemente la velocità desiderata per il velivolo, che è imposta costante.

Per semplicità lo scenario, quindi le posizioni dei target sono note a priori; per modellare la presenza di possibili pop-up, durante la simulazione, abbiamo imposto che i velivoli individuino solo quegli obiettivi che si trovano all'interno di un cerchio centrato sulla loro posizione attuale. L'ottimizzazione viene ripetuta ogni volta che un velivolo raggiunge un target. Per diminuire il tempo di calcolo necessario per l'ottimizzazione, abbiamo considerato un numero massimo di generazioni pari a 100. Questo ha portato, durante alcune simulazioni ha risultati non ottimi.

E' stata, inoltre, realizzata una interfaccia grafica per visualizzare i risultati della simulazione (vedi Figura 21).

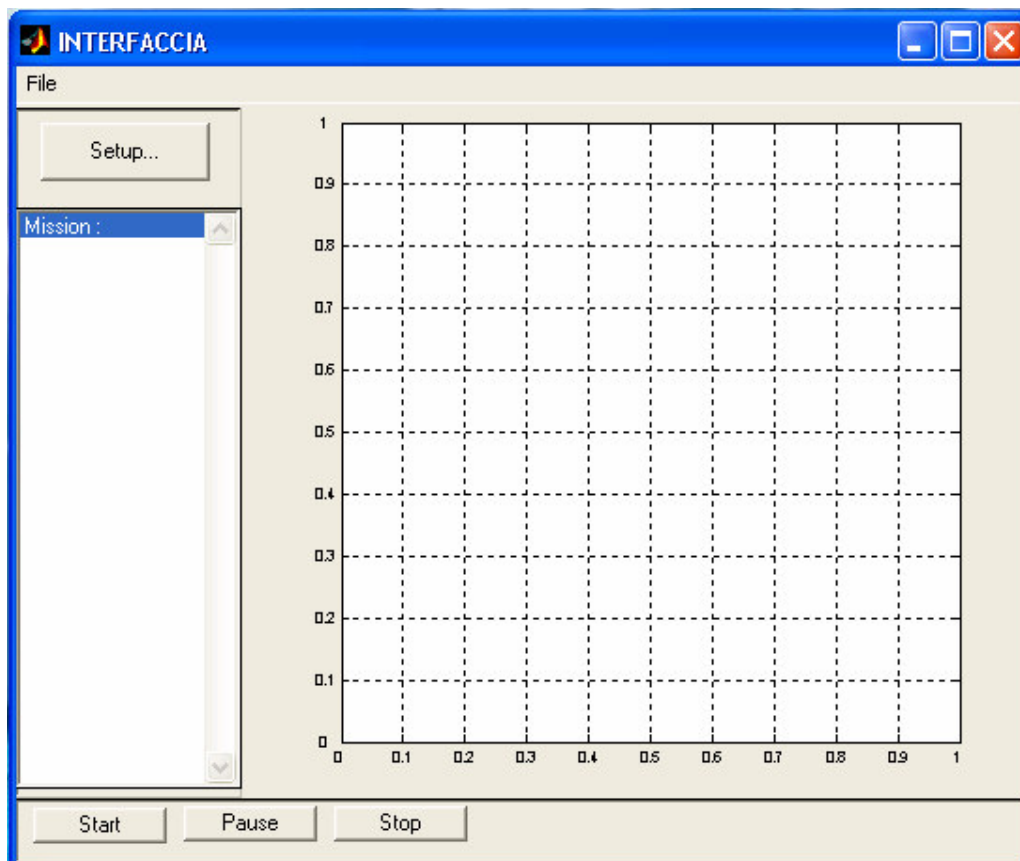


Figura 21: Interfaccia per la simulazione

L'interfaccia consente, attraverso il tasto di Setup, di settare i parametri per una nuova simulazione, quindi:

- Numero di ostacoli
- Numero di velivoli
- Numero di target
- Vettore delle capacità e degli score
- Vettore dei tempi massimi
- Inizializzare lo scenario

4.2.1 Minimizza il Tempo di Missione con Vincoli

Consideriamo lo scenario mostrato in Figura:

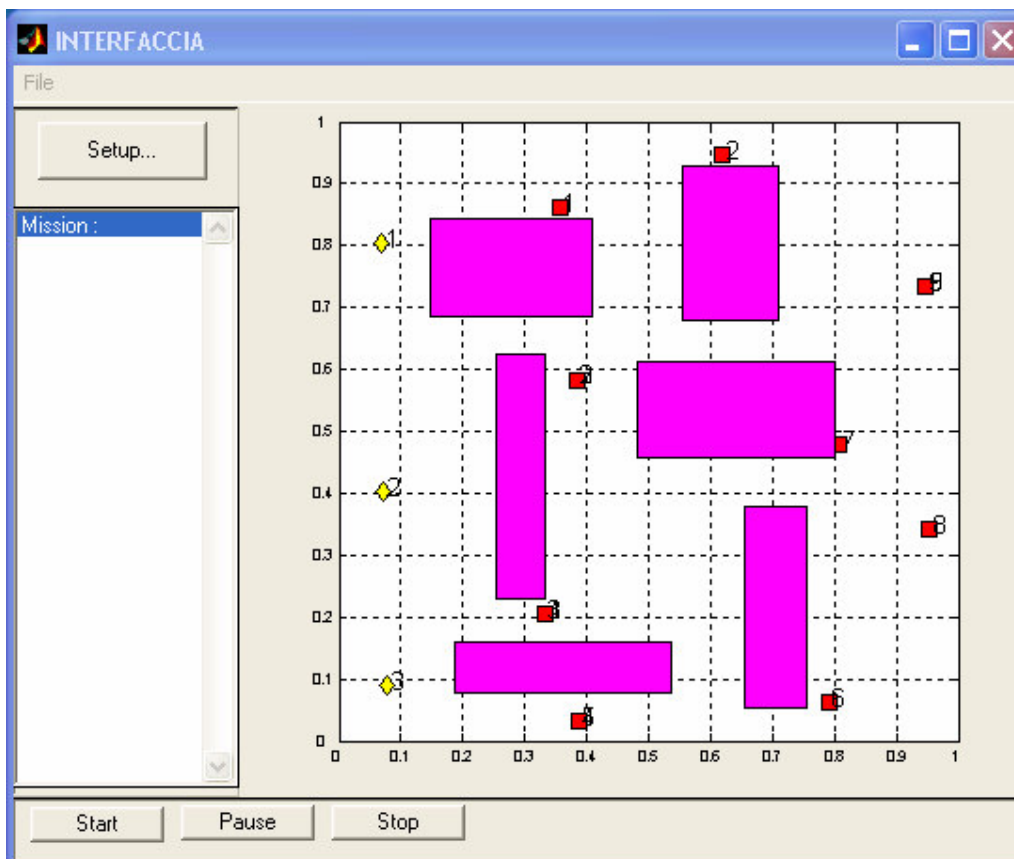


Figura 22

I tempi massimi per la simulazione risultano pari a [1.3 1.3 1.5].

Scopo della simulazione è mostrare che l'algoritmo assegna ai velivoli i target in modo da non annullare mai il tempo residuo di arrivo sul rendez-vous. Le fasi della simulazione sono mostrate in Figura 23.

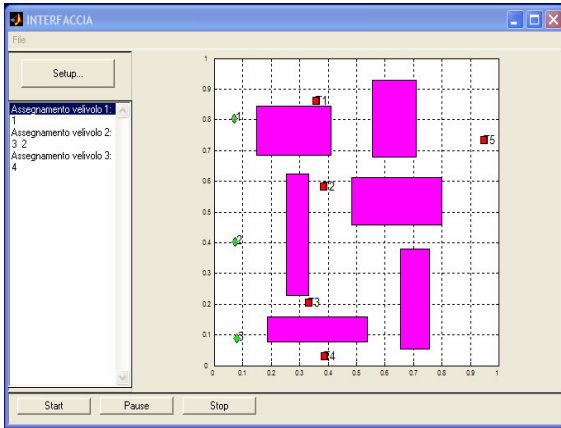


Figura 23.1: Workspace iniziale

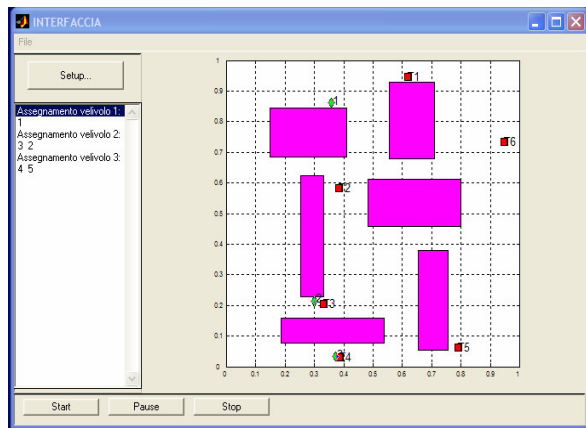


Figura23.2: Workspace 1 ottimizzazione

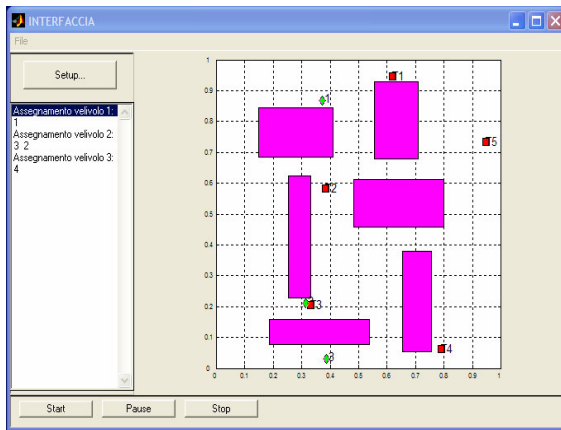


Figura 23.3: Workspace 2 ottimizzazione

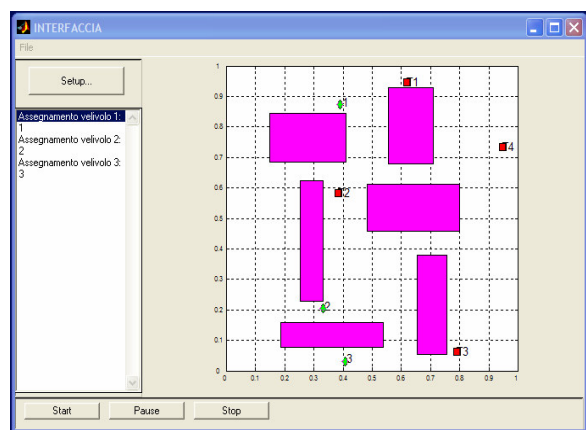


Figura23.4: Workspace 3 ottimizzazione

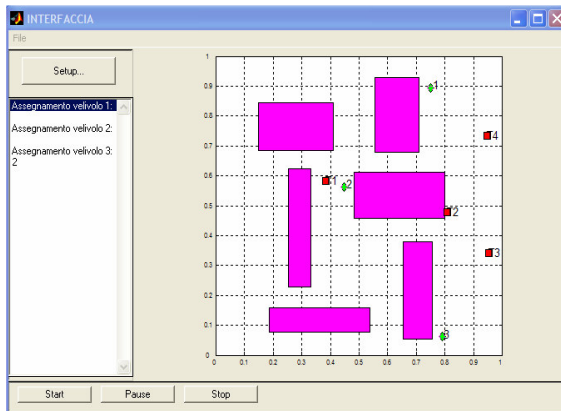


Figura 23.5: Workspace 4 ottimizzazione

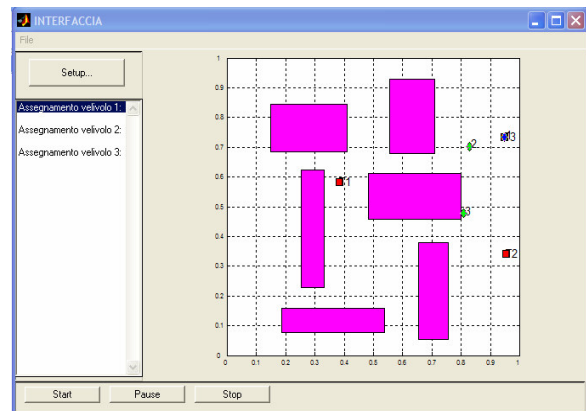


Figura23.2: Workspace 1 ottimizzazione

Figura 23: Fasi della simulazione

Alla partenza della simulazione i velivoli si trovano ad operare nell'ambiente mostrato in Figura 23.1; l'assegnamento ottimo ottenuto è il seguente:

- Velivolo 1: target 1

- Velivolo 2: target 3 -> target 2
- Velivolo 3: target 4

Il primo velivolo che raggiunge un obiettivo è il velivolo 1. A questo punto entrano in gioco due nuovi target (target 1 e 5 in Figura 23.2), riparte l'ottimizzazione che restituisce l'assegnamento:

- Velivolo 1: target 1
- Velivolo 2: target 3 -> target 2
- Velivolo 3: target 4 -> target 5

Il velivolo 3 raggiunge il target 4; il workspace è mostrato in Figura 23.3, non compaiono pop-up. L'ottimizzazione restituisce:

- Velivolo 1: target 1
- Velivolo 2: target 3 -> target 2
- Velivolo 3: target 4

Il velivolo 2 raggiunge il target 3; il workspace è mostrato in Figura 23.4, non compaiono pop-up. L'ottimizzazione restituisce:

- Velivolo 1: target 1
- Velivolo 2: target 2
- Velivolo 3: target 3

Il velivolo 1 raggiunge il target assegnatogli; il nuovo workspace è mostrato in Figura 23.5, vengono rilevati 2 nuovi obiettivi (target 2 e 3); l'ottimizzazione restituisce:

- Velivolo 1:
- Velivolo 2:

➤ Velivolo 3: target 2

L'ottimizzazione, in questo caso, non attribuisce più al velivolo 2 il target che in Figura 23.5 è indicato come target 1. Quando il velivolo 3 raggiunge i target 2, il velivolo 1 è già arrivato al rendez-vous e quindi non entra più in gioco nella simulazione; riparte l'ottimizzazione che comunque non assegna gli obiettivi rimasti ai velivoli.

In Tabella 4 sono mostrati gli andamenti dei tempi residui durante la simulazione:

Velivolo 1	Velivolo 2	Velivolo 3
1.0016	1.0016	1.2016
0.9850	0.9850	1.1851
0.9673	0.9673	1.1674
0.7265	0.7261	0.9266
0.5804	0.5804	0.7806
0.3269	0.3269	0.3645
0.3269	0.0398	0.0731

Tabella 4: Andamento dei Tempi Residui

La durata della simulazione è mediamente pari a 147 secondi.

Altre prove hanno dato diverse soluzioni, comunque coerenti con l'ipotesi che i tempi residui non si debbano annullare.

Un'ulteriore simulazione è stata svolta con Tempi Massimi diversi per i velivoli ([1.3 1.4 1.7]). In questo caso, la missione si svolge in modo identico alla precedente fino

alla 4° ottimizzazione; lo scenario è identico. L'ottimizzazione restituisce la seguente allocazione:

- Velivolo 1:
- Velivolo 2: target 1
- Velivolo 3: target 3 -> target 2

Il velivolo 2 raggiunge il target 1; l'ottimizzazione restituisce la seguente allocazione:

- Velivolo 1:
- Velivolo 2:
- Velivolo 3: target 2 (ex tg 3)-> target 1(ex tg 2)

L'ultima ottimizzazione porta il velivolo 3 a visitare il target 1; in questa missione i velivoli riescono, dunque, a coprire tutti i target, comunque i tempi residui risultano ancora diversi da zero e valgono [0.3270 0.0285 0.1667].

4.2.2 Target con Score Esteso

Consideriamo il seguente scenario, mostrato in Figura:

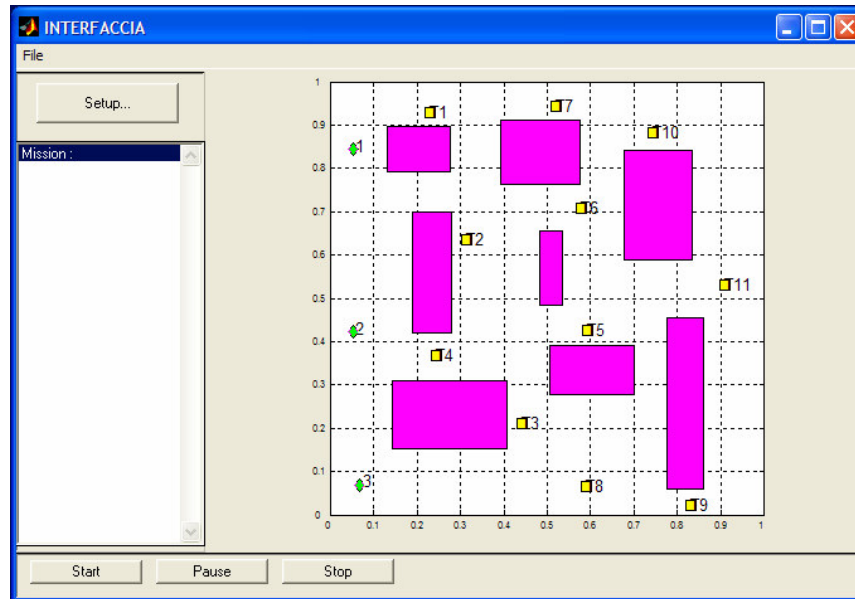


Figura 24

I parametri per la simulazione sono i seguenti:

- Capacità Operativa pari a [1 1 1]
- Gli Score dei target sono stati generati casualmente, è valgono [0.792 0.922 0.738 0.176 0.405 0.935 0.917 0.41 0.894 0.06 0.353]
- Dimensione della popolazione pari a 100
- Numero di generazioni pari a 200
- $\alpha=5$, $\beta=.5$

Scopo della simulazione è mostrare che l'algorithm assegna i target ai velivoli in modo da massimizzare ad ogni passo le capacità operative dei velivoli, e che tali

capacità non si annullano quando si arriva al rendez-vous. Le fasi della simulazione sono mostrate in Figura.

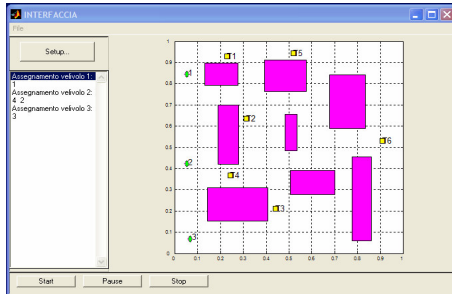


Figura 25.1: Assegnamento iniziale

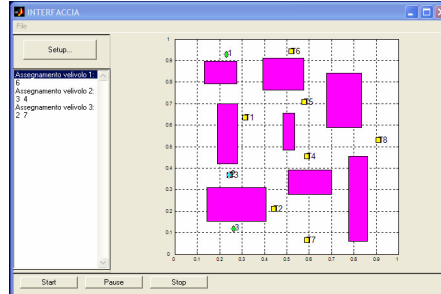


Figura 25.2: 1^o ottimizzazione

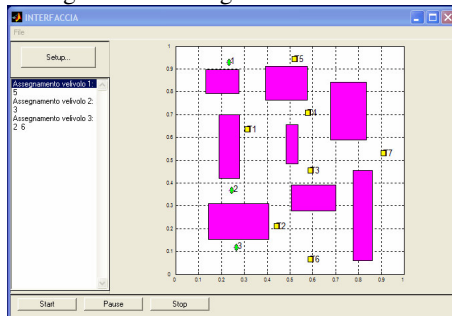


Figura 25.3: 2^o ottimizzazione

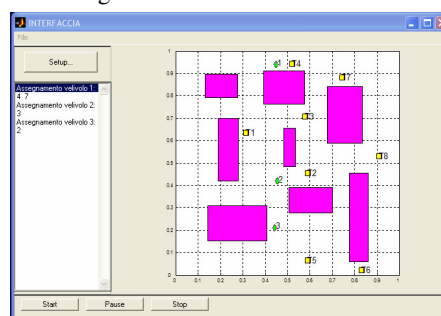


Figura 25.4: 3^o ottimizzazione

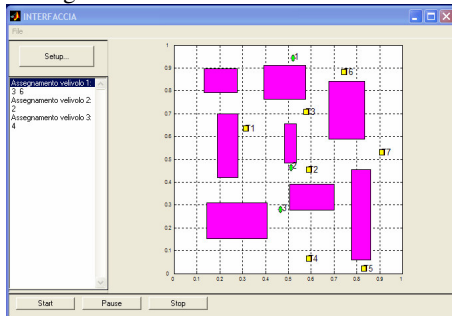


Figura 25.5: 4^o ottimizzazione

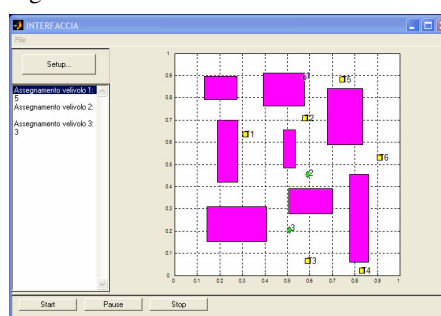


Figura 25.6: 5^o ottimizzazione

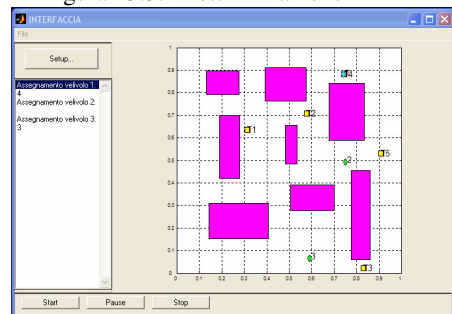


Figura 25.7: 6^o ottimizzazione

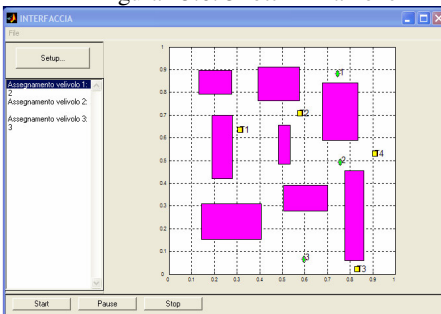


Figura 25.8: 7^o ottimizzazione

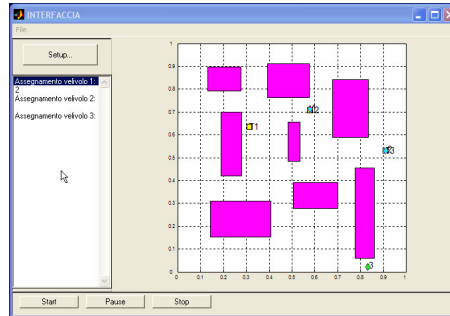


Figura 25.9: 8° ottimizzazione

Figura 25: Fasi della Simulazione

Ad ogni passo l’algoritmo massimizza le capacità residue, i risultati sono mostrati in Tabella.

Velivolo 1	Velivolo 2	Velivolo 3
0.8829	0.9984	0.9984
0.8829	0.9984	0.9984
0.8813	0.9966	0.8711
0.8617	0.9964	0.8709
0.8615	0.9963	0.8709
0.8605	0.9852	0.8288
0.8605	0.9852	0.8288
0.8585	0.9842	0.6434
0.7329	0.9842	0.4924

La distanza percorsa è molto bassa

La distanza percorsa è molto bassa

Tabella 4: Andamento delle Capacità Residue

Utilizzando valori più bassi per α , β , ad esempio $\alpha=2$ e $\beta=0.2$ otteniamo una simulazione che mostra l’effettiva “bontà” dell’algoritmo. Infatti alla 3°

ottimizzazione l’algoritmo non riesce a individuare un assegnamento che permetta ai velivoli di arrivare tutti al rendez-vous con una capacità residua maggiore di zero, la simulazione viene ripresa con gli assegnamenti precedenti, ossia:

- Velivolo 1:
- Velivolo 2:
- Velivolo 3: target 3 -> target 7

Il velivolo 2 raggiunge il secondo obiettivo assegnatogli; riparte una nuova ottimizzazione che comunque non assegna ulteriori target, e i velivoli tornano al rendez-vous.

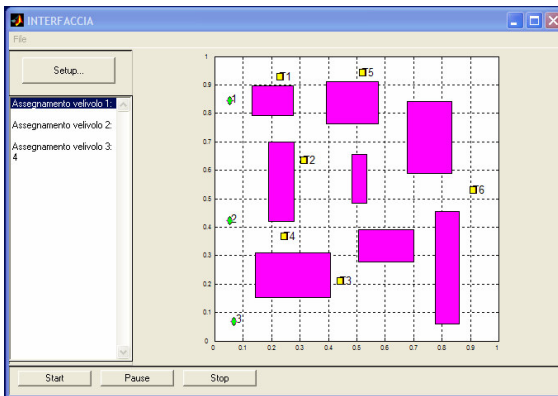


Figura 26.1 Assegnamento iniziale

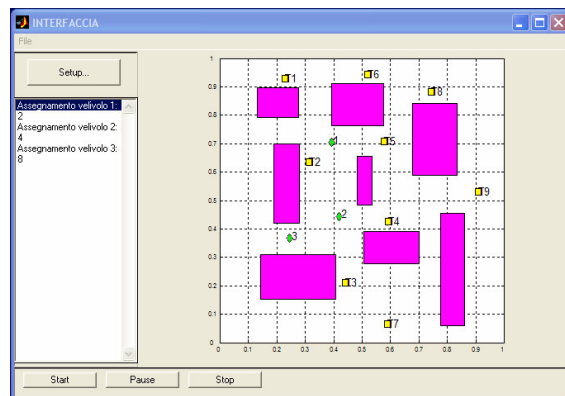


Figura 26.2 1° ottimizzazione

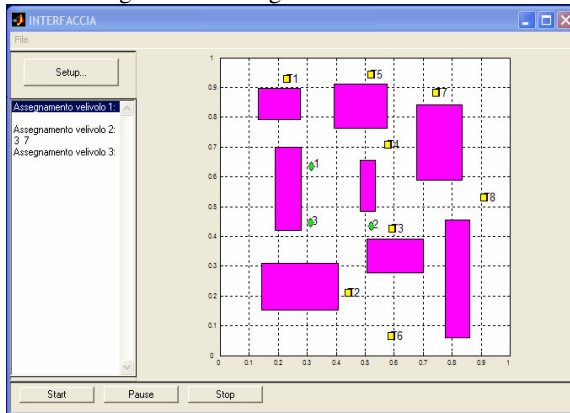


Figura 26.3 2° ottimizzazione

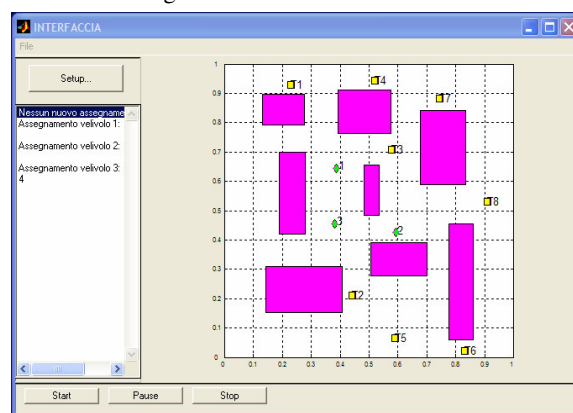


Figura 26.4 3° ottimizzazione

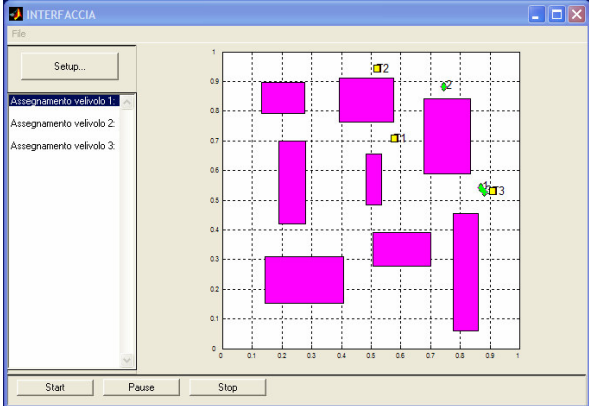


Figura 26.5 4° ottimizzazione

Figura 26: Fasi della Simulazione